

500.40255X00

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE



#4/
16-9-01

Applicant(s): WAJIMA, et al
Serial No.: Not assigned
Filed: June 20, 2001
Title: METHOD AND SYSTEM OF PROGRAM MANAGEMENT FOR
 USING COMPONENT MODEL
Group: Not assigned

LETTER CLAIMING RIGHT OF PRIORITY

Honorable Commissioner of
 Patents and Trademarks
Washington, D.C. 20231

June 20, 2001

Sir:

Under the provisions of 35 USC 119 and 37 CFR 1.55, the applicant(s) hereby claim(s) the right of priority based on Japanese Patent Application No.(s) 2001-026250, filed February 2, 2001.

A certified copy of said Japanese Application is attached.

Respectfully submitted,

ANTONELLI, TERRY, STOUT & KRAUS, LLP

Carl I. Brundidge
Registration No. 29,621

CIB/amr
Attachment
(703) 312-6600

日 本 国 特 許 庁
JAPAN PATENT OFFICE

J1036 U.S. PTO
09/884067
06/20/01

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出 願 年 月 日
Date of Application:

2 0 0 1 年 2 月 2 日

出 願 番 号
Application Number:

特 願 2 0 0 1 - 0 2 6 2 5 0

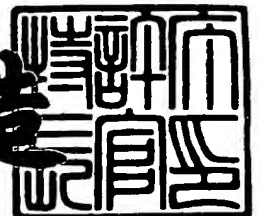
出 願 人
Applicant(s):

株式会社日立製作所

2 0 0 1 年 5 月 1 1 日

特 許 庁 長 官
Commissioner,
Japan Patent Office

及 川 耕 造



出証番号 出証特 2 0 0 1 - 3 0 3 9 4 8 0

【書類名】 特許願

【整理番号】 K00019181A

【提出日】 平成13年 2月 2日

【あて先】 特許庁長官殿

【国際特許分類】 G06F 17/00

【発明者】

【住所又は居所】 神奈川県川崎市幸区鹿島田 8 9 0 番地 株式会社日立製作所 ビジネスソリューション開発本部内

【氏名】 輪島 裕之

【発明者】

【住所又は居所】 神奈川県川崎市幸区鹿島田 8 9 0 番地 株式会社日立製作所 ビジネスソリューション開発本部内

【氏名】 土田 修己

【発明者】

【住所又は居所】 神奈川県横浜市戸塚区戸塚町 5 0 3 0 番地 株式会社日立製作所 ソフトウェア事業部内

【氏名】 佐藤 将夫

【特許出願人】

【識別番号】 000005108

【氏名又は名称】 株式会社日立製作所

【代理人】

【識別番号】 100075096

【弁理士】

【氏名又は名称】 作田 康夫

【手数料の表示】

【予納台帳番号】 013088

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

特 2 0 0 1 - 0 2 6 2 5 0

【物件名】 図面 1

【物件名】 要約書 1

【ブルーフの要否】 要

【書類名】 明細書

【発明の名称】 プログラム制御方法およびシステム並びにその処理プログラム
およびそのプログラムを格納した記録媒体

【特許請求の範囲】

【請求項 1】

インタフェース名とコンポーネント種別とコンポーネント参照情報が対応付けられて格納されているコンポーネント参照情報管理領域を参照し、入力したインタフェース名とコンポーネント種別に対応する前記格納されたコンポーネント参照情報を出力する第一のステップと、

コンポーネントを要求するリクエストであり、コンポーネント種別をパラメータとして保持するリクエストを入力し、前記リクエストからコンポーネント種別を取得し、出力する第二のステップと、

インタフェース名を保持し、前記コンポーネントを要求するリクエストを入力し、前記リクエストを前記第二のステップへ出力し、前記コンポーネント種別を該第二のステップから入力し、前記入力したコンポーネント種別と前記保持するインタフェース名を前記第一のステップへ出力し、前記コンポーネント参照情報を該第一のステップから入力して、前記入力したコンポーネント参照情報を出力する第三のステップとを有するプログラム制御方法

【請求項 2】

インタフェース名とコンポーネント種別とコンポーネント参照情報が対応付けられて格納されているコンポーネント参照情報管理領域を参照し、入力したインタフェース名とコンポーネント種別に対応する前記格納されたコンポーネント参照情報を出力する第一のステップと、

インタフェース名とコンポーネント種別とコンポーネント判定条件が対応付けられて格納されているコンポーネント種別管理領域を参照し、入力したインタフェース名とコンポーネント判定情報に対応する前記格納されたコンポーネント種別を出力する第二のステップと、

コンポーネントを要求するリクエストであり、コンポーネント判定情報をパラメータとして保持するリクエストを入力し、前記リクエストからコンポーネント

判定情報を取得し、出力する第三のステップと、

インタフェース名を保持し、前記コンポーネントを要求するリクエストを入力し、前記リクエストを前記第三のステップへ出力し、前記コンポーネント判定情報を該第三のステップから入力し、

前記入力したコンポーネント判定情報と前記保持するインタフェース名を前記第二のステップへ出力し、前記コンポーネント種別を該第二のステップから入力し、前記入力したコンポーネント種別と前記保持するインタフェース名を前記第一のステップへ出力し、前記コンポーネント参照情報を該第一のステップから入力して、前記入力したコンポーネント参照情報を出力する第三のステップとを有するプログラム制御方法

【請求項 3】

インタフェース名とコンポーネント種別とコンポーネント参照情報が対応付けられて格納されているコンポーネント参照情報管理領域を参照し、入力したインタフェース名とコンポーネント種別に対応する前記格納されたコンポーネント参照情報を出力する第一のステップと、

インタフェース名とコンポーネント種別とコンポーネント判定条件が対応付けられて格納されているコンポーネント種別管理領域を参照し、入力したインタフェース名とコンポーネント判定情報に対応する前記格納されたコンポーネント種別を出力する第二のステップと、

コンポーネント判定情報を管理し、コンポーネントを要求するリクエストを入力し、前記管理するコンポーネント判定情報を出力する第三のステップと、
インタフェース名を保持し、

前記コンポーネントを要求するリクエストを入力し、
前記リクエストを前記第三のステップへ出力し、前記コンポーネント判定情報を該第三のステップから入力し、前記入力したコンポーネント判定情報と前記保持するインタフェース名を前記第二のステップへ出力し、前記コンポーネント種別を該第二のステップから入力し、前記入力したコンポーネント種別と前記保持するインタフェース名を前記第一のステップへ出力し、前記コンポーネント参照情報を該第一のステップから入力して、前記入力したコンポーネント参照情報を出

力する第三のステップとを有するプログラム制御方法

【請求項4】

インタフェース名とコンポーネント種別とコンポーネント参照情報が対応付けられて格納されているコンポーネント参照情報管理領域を参照し、入力したインタフェース名とコンポーネント種別に対応する前記格納されたコンポーネント参照情報を出力する第一の手段と、

コンポーネントを要求するリクエストであり、コンポーネント種別をパラメータとして保持するリクエストを入力し、前記リクエストからコンポーネント種別を取得し、出力する第二の手段と、

インタフェース名を保持し、前記コンポーネントを要求するリクエストを入力し、前記リクエストを前記第二の手段へ出力し、前記コンポーネント種別を該第二の手段から入力し、前記入力したコンポーネント種別と前記保持するインタフェース名を前記第一の手段へ出力し、前記コンポーネント参照情報を該第一の手段から入力して、前記入力したコンポーネント参照情報を出力する第三の手段とを有するプログラム制御システム

【請求項5】

インタフェース名とコンポーネント種別とコンポーネント参照情報が対応付けられて格納されているコンポーネント参照情報管理領域を参照し、入力したインタフェース名とコンポーネント種別に対応する前記格納されたコンポーネント参照情報を出力する第一の手段と、

インタフェース名とコンポーネント種別とコンポーネント判定条件が対応付けられて格納されているコンポーネント種別管理領域を参照し、入力したインタフェース名とコンポーネント判定情報に対応する前記格納されたコンポーネント種別を出力する第二の手段と、

コンポーネントを要求するリクエストであり、コンポーネント判定情報をパラメータとして保持するリクエストを入力し、前記リクエストからコンポーネント判定情報を取得し、出力する第三の手段と、

インタフェース名を保持し、前記コンポーネントを要求するリクエストを入力し、前記リクエストを前記第三の手段へ出力し、前記コンポーネント判定情報を

該第三の手段から入力し、

前記入力したコンポーネント判定情報と前記保持するインタフェース名を前記第二の手段へ出力し、前記コンポーネント種別を該第二の手段から入力し、前記入力したコンポーネント種別と前記保持するインタフェース名を前記第一の手段へ出力し、前記コンポーネント参照情報を該第一の手段から入力して、前記入力したコンポーネント参照情報を出力する第三の手段とを有するプログラム制御方法

【請求項 6】

インタフェース名とコンポーネント種別とコンポーネント参照情報が対応付けられて格納されているコンポーネント参照情報管理領域を参照し、入力したインタフェース名とコンポーネント種別に対応する前記格納されたコンポーネント参照情報を出力する第一の手段と、

インタフェース名とコンポーネント種別とコンポーネント判定条件が対応付けられて格納されているコンポーネント種別管理領域を参照し、入力したインタフェース名とコンポーネント判定情報に対応する前記格納されたコンポーネント種別を出力する第二の手段と、

コンポーネント判定情報を管理し、コンポーネントを要求するリクエストを入力し、前記管理するコンポーネント判定情報を出力する第三の手段と、
インタフェース名を保持し、

前記コンポーネントを要求するリクエストを入力し、
前記リクエストを前記第三の手段へ出力し、前記コンポーネント判定情報を該第三の手段から入力し、前記入力したコンポーネント判定情報と前記保持するインタフェース名を前記第二の手段へ出力し、前記コンポーネント種別を該第二の手段から入力し、前記入力したコンポーネント種別と前記保持するインタフェース名を前記第一の手段へ出力し、前記コンポーネント参照情報を該第一の手段から入力して、前記入力したコンポーネント参照情報を出力する第三の手段とを有するプログラム制御方法

【請求項 7】

インタフェース名とコンポーネント種別とコンポーネント参照情報が対応付け

られて格納されているコンポーネント参照情報管理領域を参照し、入力したインタフェース名とコンポーネント種別に対応する前記格納されたコンポーネント参照情報を出力する第一のステップと、

コンポーネントを要求するリクエストであり、コンポーネント種別をパラメータとして保持するリクエストを入力し、前記リクエストからコンポーネント種別を取得し、出力する第二のステップと、

インタフェース名を保持し、前記コンポーネントを要求するリクエストを入力し、前記リクエストを前記第二のステップへ出力し、前記コンポーネント種別を該第二のステップから入力し、前記入力したコンポーネント種別と前記保持するインタフェース名を前記第一のステップへ出力し、前記コンポーネント参照情報を該第一のステップから入力して、前記入力したコンポーネント参照情報を出力する第三のステップとを有するプログラム制御プログラム

【請求項 8】

インタフェース名とコンポーネント種別とコンポーネント参照情報が対応付けられて格納されているコンポーネント参照情報管理領域を参照し、入力したインタフェース名とコンポーネント種別に対応する前記格納されたコンポーネント参照情報を出力する第一のステップと、

インタフェース名とコンポーネント種別とコンポーネント判定条件が対応付けられて格納されているコンポーネント種別管理領域を参照し、入力したインタフェース名とコンポーネント判定情報に対応する前記格納されたコンポーネント種別を出力する第二のステップと、

コンポーネントを要求するリクエストであり、コンポーネント判定情報をパラメータとして保持するリクエストを入力し、前記リクエストからコンポーネント判定情報を取得し、出力する第三のステップと、

インタフェース名を保持し、前記コンポーネントを要求するリクエストを入力し、前記リクエストを前記第三のステップへ出力し、前記コンポーネント判定情報を該第三のステップから入力し、

前記入力したコンポーネント判定情報と前記保持するインタフェース名を前記第二のステップへ出力し、前記コンポーネント種別を該第二のステップから入力

し、前記入力したコンポーネント種別と前記保持するインタフェース名を前記第一のステップへ出力し、前記コンポーネント参照情報を該第一のステップから入力して、前記入力したコンポーネント参照情報を出力する第三のステップとを有するプログラム制御プログラム

【請求項 9】

インタフェース名とコンポーネント種別とコンポーネント参照情報が対応付けられて格納されているコンポーネント参照情報管理領域を参照し、入力したインタフェース名とコンポーネント種別に対応する前記格納されたコンポーネント参照情報を出力する第一のステップと、

インタフェース名とコンポーネント種別とコンポーネント判定条件が対応付けられて格納されているコンポーネント種別管理領域を参照し、入力したインタフェース名とコンポーネント判定情報に対応する前記格納されたコンポーネント種別を出力する第二のステップと、

コンポーネント判定情報を管理し、コンポーネントを要求するリクエストを入力し、前記管理するコンポーネント判定情報を出力する第三のステップと、
インタフェース名を保持し、

前記コンポーネントを要求するリクエストを入力し、
前記リクエストを前記第三のステップへ出力し、前記コンポーネント判定情報を該第三のステップから入力し、前記入力したコンポーネント判定情報と前記保持するインタフェース名を前記第二のステップへ出力し、前記コンポーネント種別を該第二のステップから入力し、前記入力したコンポーネント種別と前記保持するインタフェース名を前記第一のステップへ出力し、前記コンポーネント参照情報を該第一のステップから入力して、前記入力したコンポーネント参照情報を出力する第三のステップとを有するプログラム制御プログラム

【請求項 10】

インタフェース名とコンポーネント種別とコンポーネント参照情報が対応付けられて格納されているコンポーネント参照情報管理領域を参照し、入力したインタフェース名とコンポーネント種別に対応する前記格納されたコンポーネント参照情報を出力する第一のステップと、

コンポーネントを要求するリクエストであり、コンポーネント種別をパラメータとして保持するリクエストを入力し、前記リクエストからコンポーネント種別を取得し、出力する第二のステップと、

インタフェース名を保持し、前記コンポーネントを要求するリクエストを入力し、前記リクエストを前記第二のステップへ出力し、前記コンポーネント種別を該第二のステップから入力し、前記入力したコンポーネント種別と前記保持するインタフェース名を前記第一のステップへ出力し、前記コンポーネント参照情報を該第一のステップから入力して、前記入力したコンポーネント参照情報を出力する第三のステップとを有するプログラム制御プログラムを格納した計算機読み取り可能な記録媒体

【請求項 11】

インタフェース名とコンポーネント種別とコンポーネント参照情報が対応付けられて格納されているコンポーネント参照情報管理領域を参照し、入力したインタフェース名とコンポーネント種別に対応する前記格納されたコンポーネント参照情報を出力する第一のステップと、

インタフェース名とコンポーネント種別とコンポーネント判定条件が対応付けられて格納されているコンポーネント種別管理領域を参照し、入力したインタフェース名とコンポーネント判定情報に対応する前記格納されたコンポーネント種別を出力する第二のステップと、

コンポーネントを要求するリクエストであり、コンポーネント判定情報をパラメータとして保持するリクエストを入力し、前記リクエストからコンポーネント判定情報を取得し、出力する第三のステップと、

インタフェース名を保持し、前記コンポーネントを要求するリクエストを入力し、前記リクエストを前記第三のステップへ出力し、前記コンポーネント判定情報を該第三のステップから入力し、

前記入力したコンポーネント判定情報と前記保持するインタフェース名を前記第二のステップへ出力し、前記コンポーネント種別を該第二のステップから入力し、前記入力したコンポーネント種別と前記保持するインタフェース名を前記第一のステップへ出力し、前記コンポーネント参照情報を該第一のステップから入

力して、前記入力したコンポーネント参照情報を出力する第三のステップとを有するプログラム制御プログラムを格納した計算機読み取り可能な記録媒体

【請求項 1 2】

インタフェース名とコンポーネント種別とコンポーネント参照情報が対応付けられて格納されているコンポーネント参照情報管理領域を参照し、入力したインタフェース名とコンポーネント種別に対応する前記格納されたコンポーネント参照情報を出力する第一のステップと、

インタフェース名とコンポーネント種別とコンポーネント判定条件が対応付けられて格納されているコンポーネント種別管理領域を参照し、入力したインタフェース名とコンポーネント判定情報に対応する前記格納されたコンポーネント種別を出力する第二のステップと、

コンポーネント判定情報を管理し、コンポーネントを要求するリクエストを入力し、前記管理するコンポーネント判定情報を出力する第三のステップと、
インタフェース名を保持し、

前記コンポーネントを要求するリクエストを入力し、
前記リクエストを前記第三のステップへ出力し、前記コンポーネント判定情報を該第三のステップから入力し、前記入力したコンポーネント判定情報と前記保持するインタフェース名を前記第二のステップへ出力し、前記コンポーネント種別を該第二のステップから入力し、前記入力したコンポーネント種別と前記保持するインタフェース名を前記第一のステップへ出力し、前記コンポーネント参照情報を該第一のステップから入力して、前記入力したコンポーネント参照情報を出力する第三のステップとを有するプログラム制御プログラムを格納した計算機読み取り可能な記録媒体

【発明の詳細な説明】

【 0 0 0 1】

【発明の属する技術分野】

本発明は、コンポーネントモデルに基づく分散オブジェクト技術に関する。

【 0 0 0 2】

【従来の技術】

分散オブジェクト技術を利用して構築した情報処理システムを分散オブジェクトシステムと呼ぶ。

【 0 0 0 3 】

分散オブジェクトシステムにおいて、再利用を可能にするため、一定の規則に従って実装されたオブジェクトをコンポーネントと呼ぶ。コンポーネントは、再利用が可能のため、流通部品として販売、購入することが可能である。コンポーネントを再利用してシステムを構築する方式をコンポーネントモデルと呼ぶ。

【 0 0 0 4 】

コンポーネントモデルに基づくシステムでは、サービスを提供するプログラムがコンポーネントを起動し、サービスを利用するプログラムがコンポーネントを呼び出すことにより望みの処理を実現する。ここで、サービスを提供するプログラムをサービス提供プログラム、サービスを利用するプログラムをサービス利用プログラムと呼ぶ。

【 0 0 0 5 】

コンポーネントモデルに基づくシステムでは、サービス利用プログラムがコンポーネントを呼び出すための識別子を取得する方法が問題になる。ここで、オブジェクトやコンポーネントを呼び出すための識別子を参照情報と呼ぶ。また、サービス利用プログラムが、コンポーネントの参照情報を取得するために利用するオブジェクトをホームと呼ぶ。

【 0 0 0 6 】

第一の公知例は、例えば、米サンマイクロシステムズ社発行の「Enterprise JavaBeans Specification, v1.1(1999)」に記載されている（Enterprise JavaBeansは米サンマイクロシステムズ社の登録商標である）。本公知例では、コンポーネントをEnterprise Beanと呼び、ホームをEJBHomeと呼ぶ。

【 0 0 0 7 】

第二の公知例は、例えば、米の標準化団体Object Management Group発行の「CORBA Components - Volume 1: Joint Revised Submission, (Aug. 2, 2000)」に記載されている。ここでCORBA(Common Object Request Broker Architecture)は、Object Management Group発行の「The Common Object Request Broker: Archi

itecture and Specification, Revision 2.0(1996)」に記載されている分散オブジェクト技術を指す（CORBAは、Object Management Groupの登録商標である）。

【0008】

ここで、従来の技術において、サービス利用プログラムがコンポーネントを呼び出す手順を、図2を用いて説明する。従来の技術では、ホーム111とコンポーネント112は一対一に対応する。ホーム111では、コンポーネント112の参照情報であるコンポーネント参照情報143を保持する。また、オブジェクト名1131と参照情報132の対応を管理するプログラムであるネーミングサービス120を設ける。

【0009】

サービス利用プログラム100は、まず呼び出したいコンポーネント112（例えば、通常顧客コンポーネント）に対応するホーム111（例えば、通常顧客ホーム）のオブジェクト名131（例えば、通常顧客）をネーミングサービス120に指定して、ホーム111（例えば、通常顧客ホーム）の参照情報132を取得する。次に、サービス利用プログラム100は、取得した参照情報132を利用してホーム111（例えば、通常顧客ホーム）を呼び出し、コンポーネント参照情報143を取得する。そして、コンポーネント参照情報143を利用してコンポーネント112（例えば、通常顧客コンポーネント）を呼び出す。

【0010】

このように、従来技術では、コンポーネント112、ホーム111、ネーミングサービス120に登録するオブジェクト名1131が全て一対一に対応している。従って、サービス利用プログラム100において、呼び出すコンポーネント112の種類を一意に識別することを可能にするためには、システムの運用管理者が、全種類のコンポーネント112に対応するオブジェクト名1131を一意に管理して、ネーミングサービス120に登録しておく必要がある。

【0011】

【発明が解決しようとする課題】

今後、コンポーネントモデルの普及に伴い、コンポーネントモデルに基づく分散オブジェクトシステムの大規模化、提供するサービスの多様化が進行すると考えられる。また、インターネットサービスの拡大、競争激化に伴い、サービス変

更の頻度も増加すると予想される。

【 0 0 1 2 】

このとき、先に述べた公知例には、以下のような課題が存在する。

コンポーネントモデルでは、多くの場合、提供するサービスごとに種類の異なるコンポーネントを提供する。コンポーネントと、ネーミングサービスに登録されるオブジェクト名は一対一に対応するため、サービスが多様化すると、ネーミングサービスに登録されるオブジェクト名の数も増加する。このとき、システムの運用管理者は、サービスの種類数すなわちコンポーネントの種類数と同数のオブジェクト名を一意に管理する必要があるため、その管理コストが増加する。また、サービス変更の頻度が増加すると、コンポーネントの変更、追加の頻度が増加する。従って、ネーミングサービスに登録されている名称を変更、追加する頻度も増加し、運用管理者がネーミングサービスの登録名称を管理するコストが増加する。

【 0 0 1 3 】

本発明の目的は、好適なコンポーネント名前管理を実現することができるプログラム制御方法およびシステムを提供することにある。

【 0 0 1 4 】

【課題を解決するための手段】

本発明は、コンポーネントモデルに基づくシステムにおいて、インタフェース名と、このインタフェースを満たす複数種類のコンポーネントを識別する情報であるコンポーネント種別と、コンポーネント参照情報の対応を管理する領域であるコンポーネント参照情報管理領域を保持することを特徴とする。

【 0 0 1 5 】

さらに本発明は、コンポーネント参照情報管理領域を参照し、インタフェース名とコンポーネント種別を入力として、対応するコンポーネント参照情報を出力するプログラムであるコンポーネント参照管理部を保持することを特徴とする。

【 0 0 1 6 】

さらに本発明は、コンポーネントのインタフェース名を一つ保持し、コンポーネント種別をパラメータとして保持するコンポーネント参照情報リクエスト電文

を入力として受け付け、コンポーネント参照情報リクエスト電文からコンポーネント種別を取得し、このコンポーネント種別と保持するインタフェース名を入力としてコンポーネント参照管理部を呼び出してコンポーネント参照情報を取得し、出力するプログラムであるホームを保持することを特徴とする。

【 0 0 1 7 】

本発明によれば、一つのホームが一つのインタフェースに対応する複数種類のコンポーネントを管理するので、起動するコンポーネントの種類が増加しても、ホームの増加を抑制できる。ホームとネーミングサービスに登録するオブジェクト名は一対一に対応するため、ネーミングサービスに登録するオブジェクト名の数も削減できる。従って、運用管理者が、システム全体で使用するオブジェクト名の一意性を容易に管理でき、その管理コストを削減できる。さらに、コンポーネントの変更、追加が発生しても、コンポーネント参照情報管理領域の変更を行なえば、既存のホームから参照情報を取得できるため、ネーミングサービスの登録情報を管理する必要がなくなり、運用管理者によるネーミングサービスの管理コストを削減できる。

【 0 0 1 8 】

【発明の実施の形態】

本発明の実施例について図面を用いて説明する。

【 0 0 1 9 】

まず、第一の実施例について図1、図3を用いて説明する。

本実施例は、サービスが多様化したときに、ネーミングサービスを管理するコストが増大するという課題を解決するものである。また、サービスが多様化すると、ホームが増加するため、ホームによるメモリ消費量が増大するという課題、さらに、ネーミングサービスに登録するオブジェクト名が増加するためネーミングサービスにおけるメモリ消費量が増大するという課題を解決するものである。

【 0 0 2 0 】

図1は本実施例の動作原理を示す。まず、サービス利用プログラム100は、ネーミングサービス120にオブジェクト名131を渡し、ホーム111の参照情報132を取得する。このときネーミングサービス120は、オブジェクト名1131と参照情報132の

対応の一覧を示す名前情報一覧130を参照する。次に、サービス利用プログラム100は、取得した参照情報132を利用してホーム111を呼び出し、コンポーネント参照情報143を取得する。ホーム111は、インタフェース名141を保持する。コンポーネント参照情報143は、コンポーネント112を呼び出すための識別子である。本実施例では、サービス利用プログラム100がホーム111を呼び出すとき、コンポーネント種別142（例えば、通常顧客）を渡すものとする。このときホーム111は、インタフェース名141とコンポーネント種別142とコンポーネント参照情報143の対応を管理するコンポーネント参照情報一覧140を参照し、保持するインタフェース名141（たとえば、顧客IF）と渡されたコンポーネント種別142に対応するコンポーネント参照情報143を返す。そして、サービス利用プログラム100は、取得したコンポーネント参照情報143を利用して、コンポーネント112（例えば、通常顧客コンポーネント）を呼び出す。

【 0 0 2 1 】

図3は本実施例の構成と処理の流れを示す。

まず、本実施例の構成を説明する。

リクエスト発行装置300は、ホーム参照情報リクエスト電文330とコンポーネント参照情報リクエスト電文340、リクエスト電文350を発行する装置であり、CPU 301、通信ネットワークコントローラ302、メモリ303で構成される。メモリ303の上にはサービス利用プログラム100が存在する。サービス利用プログラム100は、ホーム参照情報リクエスト発行制御部304、コンポーネント参照情報リクエスト発行制御部305、リクエスト発行制御部306で構成される。また、リクエスト発行装置300は通信ネットワークコントローラ302により通信ネットワーク360に接続される。

【 0 0 2 2 】

名前管理装置310は、オブジェクト名1131と参照情報132の対応を管理する装置で、CPU 301、通信ネットワークコントローラ302、メモリ303で構成される。メモリ303の上には、ネーミングサービス120と名前情報管理領域311を保持する。名前情報管理領域311は、名前情報一覧130を格納する領域である。また、名前管理装置310は、通信ネットワークコントローラ302により通信ネットワーク360に

接続される。

【 0 0 2 3 】

リクエスト処理装置320は、CPU 301、通信ネットワークコントローラ302、メモリ303で構成される。メモリ303の上には、サービス提供プログラム110が存在する。サービス提供プログラム110は、ホーム111、コンポーネント112、コンポーネント起動部323、コンポーネント参照情報管理領域322で構成される。コンポーネント参照情報管理領域322は、コンポーネント参照情報一覧140を格納する。また、リクエスト処理装置320は、通信ネットワークコントローラ302により通信ネットワーク360に接続される。

【 0 0 2 4 】

次に、本実施例の処理の流れを説明する。

サービス利用プログラム100は、ネーミングサービス120から参照情報を取得するため、ホーム参照情報リクエスト発行制御部304を用いてホーム参照情報リクエスト電文330を生成し、名前管理装置310に送信する。

【 0 0 2 5 】

名前管理装置310では、ネーミングサービス120がホーム参照情報リクエスト電文330を受信する。ホーム参照情報リクエスト電文330を構成するパラメータには、オブジェクト名131が含まれる。ネーミングサービス120は、名前情報管理領域311を参照して、受信したリクエストに含まれるオブジェクト名131に対応する参照情報132を取得し、サービス利用プログラム100に返送する。

【 0 0 2 6 】

サービス利用プログラム100は、ネーミングサービス120から参照情報132を取得すると、ホーム111からコンポーネント参照情報143を取得するため、コンポーネント参照情報リクエスト発行制御部305を用いてコンポーネント参照情報リクエスト電文340を生成し、リクエスト処理装置320に送信する。

【 0 0 2 7 】

リクエスト処理装置320では、ホーム111がコンポーネント参照情報リクエスト電文340を受信する。ホーム111は、コンポーネント参照情報管理部321を保持し、インタフェース名141を一つ保持する。コンポーネント参照情報リクエスト電

文340は各種のパラメータで構成される。本実施例では、パラメータとしてコンポーネント種別142を保持するものとする。ホーム111は、まず、受信したコンポーネント参照情報リクエスト電文340からコンポーネント種別142を取得する。次に、ホーム111は、保持するインタフェース名141と取得したコンポーネント種別142を入力としてコンポーネント参照情報管理部321を呼び出し、対応するコンポーネント参照情報143を取得する。このとき、コンポーネント参照情報管理部321は、コンポーネント参照情報管理領域322を参照し、渡されたインタフェース名141とコンポーネント種別142に対応するコンポーネント参照情報143を出力する。コンポーネント参照情報143を取得すると、ホーム111は、これをリクエスト発行装置300に返送する。

【 0 0 2 8 】

サービス利用プログラム100は、ホーム111からコンポーネント参照情報143を取得すると、コンポーネント112を呼び出すため、リクエスト発行制御部306を用いてリクエスト電文350を生成して、リクエスト処理装置320に送信する。

【 0 0 2 9 】

リクエスト処理装置320では、コンポーネント起動部323がリクエスト電文350を受信する。リクエスト電文350は各種のパラメータで構成され、パラメータの一つとしてコンポーネント参照情報143を保持する。コンポーネント起動部323は、リクエスト電文350からコンポーネント参照情報143を取得し、対応するコンポーネント112を呼び出す。

【 0 0 3 0 】

このように、本実施例によれば、一つのホームで一つのインタフェースに対応する複数種類のコンポーネントを管理するので、コンポーネントの種類が増加しても、ホームの数の増加を抑制できる。ここで、ホームとネーミングサービスで管理すべきオブジェクト名は一対一に対応するため、ネーミングサービスで管理すべきオブジェクト名を削減できる。従って、運用管理者がシステム全体で使用するオブジェクト名の一意性を容易に管理でき、ネーミングサービスの管理コストを削減できる。さらに、コンポーネントの変更、追加が発生しても、コンポーネント参照情報一覧の変更により、既存のホームから新たなコンポーネント

参照情報を取得できるため、名前情報一覧を変更する必要がなくなり、運用管理者によるネーミングサービスの管理コストを削減できる。

【 0 0 3 1 】

また、ホームの増加を抑制できるため、リクエスト処理装置における、ホームによるメモリの使用量を削減できる。さらに、ネーミングサービスにはホームの数と等しい数のオブジェクト名を登録する必要があるが、ホームの数を抑制することで、ネーミングサービスに登録するオブジェクト名を抑制できるため、名前情報一覧のエントリ数を削減し、名前管理装置におけるメモリ使用量を削減できる。

【 0 0 3 2 】

次に、本発明の第二の実施例について、図4、図5を用いて説明する。

第一の実施例では、コンポーネント参照情報を取得するとき、サービス利用プログラムにおいて、コンポーネント種別を明示的に指定していた。しかし、企業情報システムなどにおいては、コンポーネント種別はサービスの内容を示すものであり、企業戦略の観点から、これをサービス提供プログラムの外部（例えば、サービス利用プログラム）に露出させることを避けたい場合がある。本実施例は、一つのホームで複数種類のコンポーネントを管理する場合に、コンポーネント種別がサービス提供プログラムの外部に露出してしまうという課題を解決するものである。

【 0 0 3 3 】

図4は、本実施例の動作原理を示す。

本実施例では、インタフェース名2141、コンポーネント種別142、コンポーネント判定条件401の対応をコンポーネント種別一覧400として管理する。サービス利用プログラム100は、ホーム111にコンポーネント参照情報142を要求するとき、コンポーネント種別142の判定材料となる情報であるコンポーネント判定情報402を渡す。コンポーネント判定情報402は、例えば、ユーザ情報（ユーザの識別子、ユーザのロール、権限を示す情報、ユーザの所在地、使用する言語を示す情報、ユーザが利用する端末の種類など）である。ホーム111では、まず、受け取ったコンポーネント判定情報402（例えば、顧客ID=45）と、保持するインタフェ

ース名141（例えば、顧客IF）をもとに、コンポーネント種別一覧400を参照して、コンポーネント判定条件401（例えば、顧客ID < 1000）に該当するコンポーネント種別142（例えば、優待顧客）を取得する。次に、保持するインタフェース名141（例えば、顧客IF）と、取得したコンポーネント種別142（例えば、優待顧客）をもとに、コンポーネント参照情報一覧140を参照して、対応するコンポーネント参照情報143（例えば、優待顧客コンポーネント）を取得し、サービス利用プログラム100に返す。

【 0 0 3 4 】

図5は、本実施例の構成と処理の手順を示す。

本実施例では、サービス提供プログラム110の構成要素として、コンポーネント種別一覧400を管理する領域であるコンポーネント種別管理領域500が存在する。また、サービス提供プログラムの構成要素であるホーム111は、コンポーネント参照情報管理部321とコンポーネント種別判定部501で構成され、コンポーネントのインタフェース名141を一つ保持する。

【 0 0 3 5 】

サービス利用プログラム100は、ホーム111からコンポーネント参照情報143を取得するとき、コンポーネント参照情報リクエスト発行制御部305を用いてコンポーネント参照情報リクエスト電文340を生成し、リクエスト処理装置320に送信する。コンポーネント参照情報リクエスト電文340は、各種のパラメータで構成される。本実施例では、パラメータとして、コンポーネント判定情報402を保持する。

【 0 0 3 6 】

リクエスト処理装置320では、ホーム111がコンポーネント参照情報リクエスト電文340を受信する。ホーム111は、まず、受信したコンポーネント参照情報リクエスト電文340からコンポーネント判定情報402を取得する。そして、コンポーネント種別判定部501に、保持するインタフェース名141と、取得したコンポーネント判定情報402を渡して、コンポーネント種別142を取得する。このとき、コンポーネント種別判定部501は、コンポーネント種別管理領域500を参照し、渡されたインタフェース名141とコンポーネント判定情報402をもとに、対応するコンポー

ネント種別142を取得し、出力する。次に、ホーム111は、コンポーネント参照情報管理部321に、保持するインタフェース名141と取得したコンポーネント種別142を渡して、コンポーネント参照情報143を取得する。このとき、コンポーネント参照情報管理部321は、コンポーネント参照管理領域322を参照して、渡されたインタフェース名141とコンポーネント種別142に対応するコンポーネント参照情報143を取得し、出力する。そして、ホーム111は、取得したコンポーネント参照情報143をリクエスト発行装置300に返送する。

【 0 0 3 7 】

このように本実施例によれば、コンポーネント種別をサービス提供プログラムの外部に露出させずに済む。

また、コンポーネント判定情報としてユーザ情報を利用することにより、ホームにおいてユーザに応じたコンポーネントを選択し、サービスのカスタマイズを実現できる。

【 0 0 3 8 】

次に、本発明の第三の実施例について、図6を用いて説明する。

第一の実施例、第二の実施例では、ホームにおいて、受け取ったコンポーネント参照情報リクエスト電文に対応するコンポーネントを判定するとき、サービス利用プログラムから渡された情報（コンポーネント種別や、コンポーネント判定情報）を判定材料として利用していた。従って、コンポーネント参照情報リクエスト電文に、判定の材料となる情報が含まれなければ、複数種類のコンポーネントから対応するコンポーネントを判定できない。

【 0 0 3 9 】

本実施例は、コンポーネント参照情報リクエスト電文に判定材料となる情報が含まれない構成では、一つのホームで一種類のコンポーネントしか管理できないため、コンポーネントの増加に伴ってホームも増加し、ホームによるメモリ消費量が増加するという課題を解決するものである。

【 0 0 4 0 】

図6は、本実施例の動作原理を示す。

本実施例では、ホーム111がコンポーネント判定情報402を管理する。コンポーネント判定情報402は、例えば日付や時刻などである。ホーム111は、サービス利用プログラム100にコンポーネント参照情報143を返すとき、コンポーネント種別一覧400を参照し、保持するインタフェース名141（例えば、顧客IF）と管理するコンポーネント判定情報402（例えば、日付=3/Jan/2001）をもとに、コンポーネント判定条件401（例えば、1/Jan/2000 ≤ 日付）に該当するコンポーネント種別142（例えば顧客V2）を取得する。そして、コンポーネント参照情報一覧140を参照し、保持するインタフェース名141（例えば、顧客IF）と、取得したコンポーネント種別142（例えば、顧客V2）に対応するコンポーネント参照情報143（例えば、顧客V2コンポーネント）を取得して、サービス利用プログラム100に返す。

【0041】

このように本実施例によれば、コンポーネント参照情報リクエスト電文にコンポーネント判定情報が含まれないような構成でも、一つのホームで複数種類のコンポーネントを管理し、ホームによるメモリ消費量を削減できる。

【0042】

また、コンポーネント判定情報として、日付や日時を利用することにより、コンポーネントのバージョンアップを行う場合に、サービス利用プログラムの変更を伴わずに、さらに、サービス提供プログラムの再起動を伴わずに、自動的にコンポーネントの入れ替えを実現できる。

【0043】

以上により、一つのホームが、一つのインタフェースに対応する複数種類のコンポーネントを管理するので、起動するコンポーネントの種類が増加しても、ホームの増加を抑制できる。ホームとネーミングサービスに登録するオブジェクト名は一対一に対応するため、ネーミングサービスに登録するオブジェクト名の数も削減できる。従って、運用管理者が、システム全体で使用するオブジェクト名の一意性を容易に管理でき、その管理コストを削減できる。さらに、コンポーネントの変更、追加が発生しても、コンポーネント参照情報管理領域の変更を行なえば、既存のホームから参照情報を取得できるため、ネーミングサービスの登録

情報を管理する必要がなくなり、運用管理者によるネーミングサービスの管理コストを削減できる。

【 0 0 4 4 】

【発明の効果】

本発明によれば、好適なコンポーネント名前管理を実現することができる。

【図面の簡単な説明】

【図1】

本発明の動作原理を説明する図である。

【図2】

従来技術の動作原理を説明する図である。

【図3】

第一の実施例の構成と処理の流れを説明する図である。

【図4】

第二の実施例の動作原理を説明する図である。

【図5】

第二の実施例の構成と処理の流れを説明する図である。

【図6】

第三の実施例の動作原理を説明する図である。

【符号の説明】

- 100 … サービス利用プログラム
- 110 … サービス提供プログラム
- 111 … ホーム
- 112 … コンポーネント
- 120 … ネーミングサービス
- 130 … 名前情報一覧
- 131 … オブジェクト名
- 132 … 参照情報
- 140 … コンポーネント参照情報一覧

141 … インタフェース名

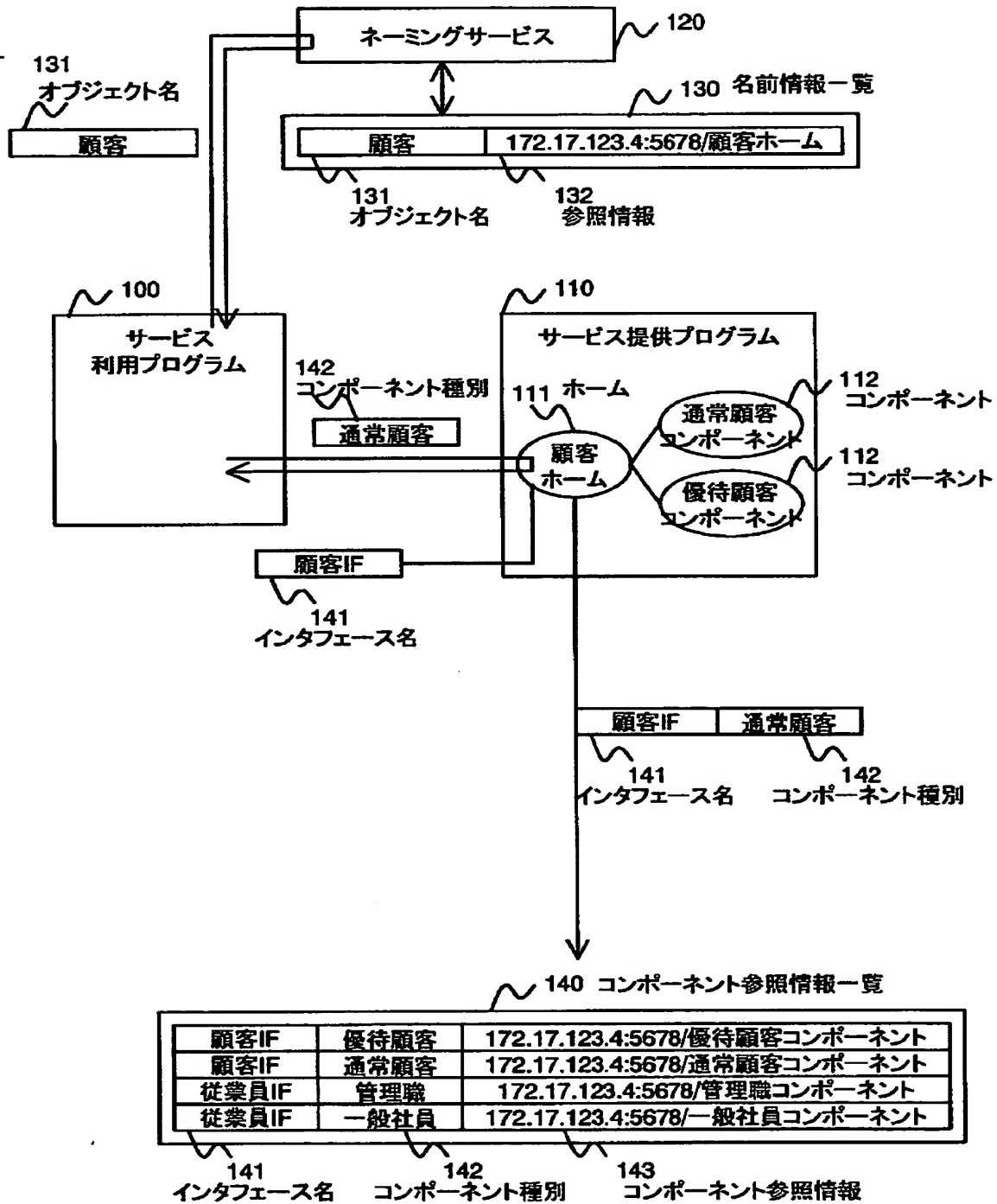
142 … コンポーネント種別

143 … コンポーネント参照情報

【書類名】 図面

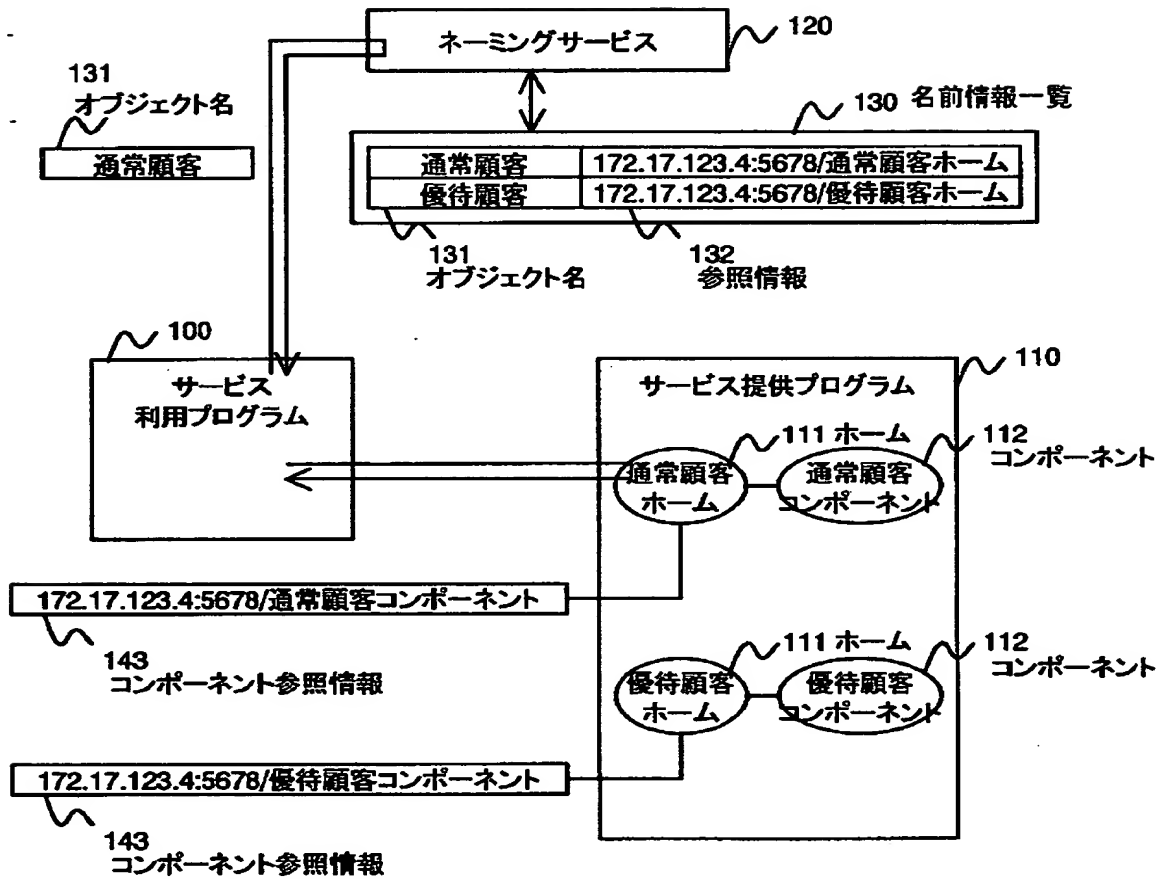
【図 1】

図 1

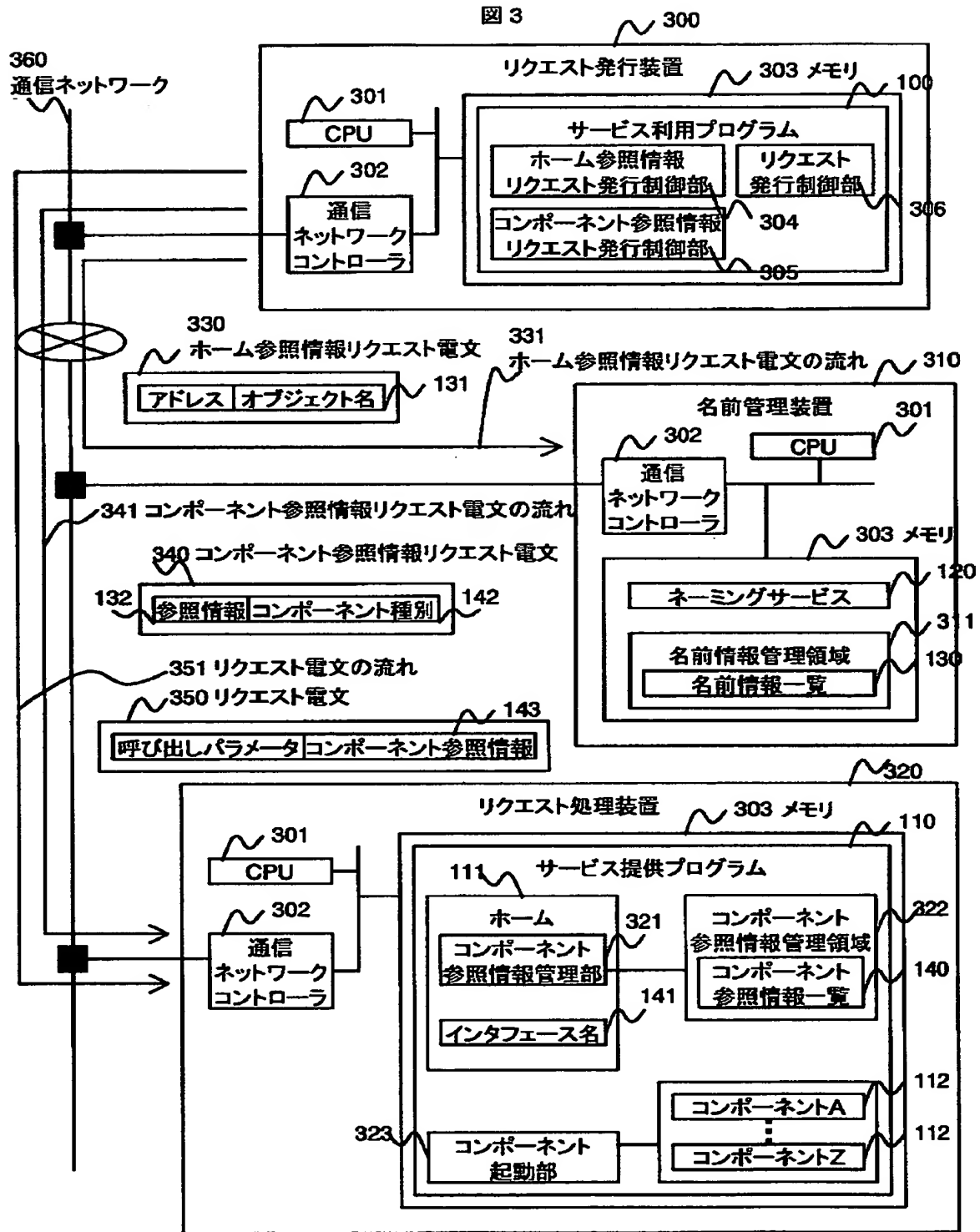


【図 2】

図 2

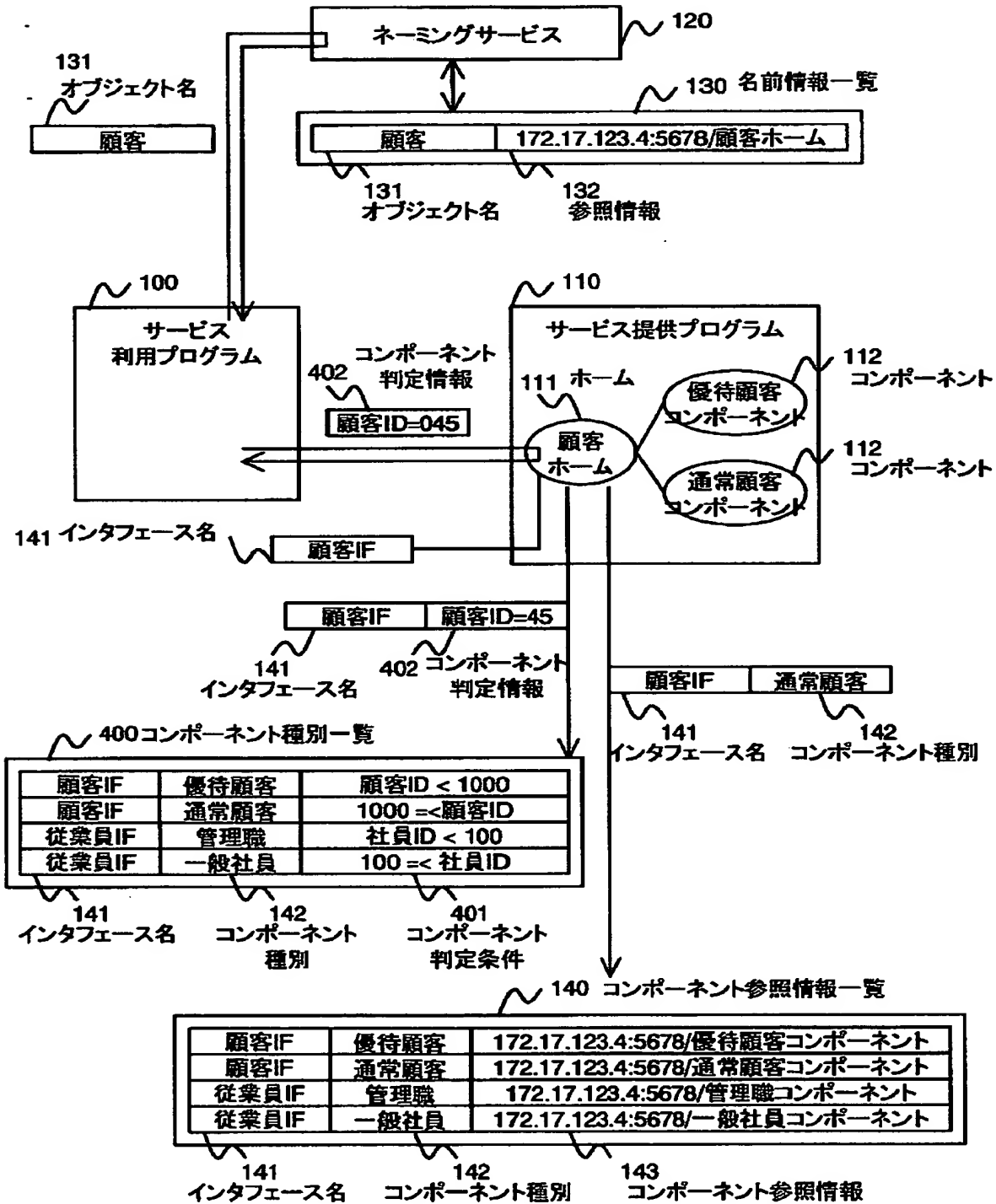


【図 3】

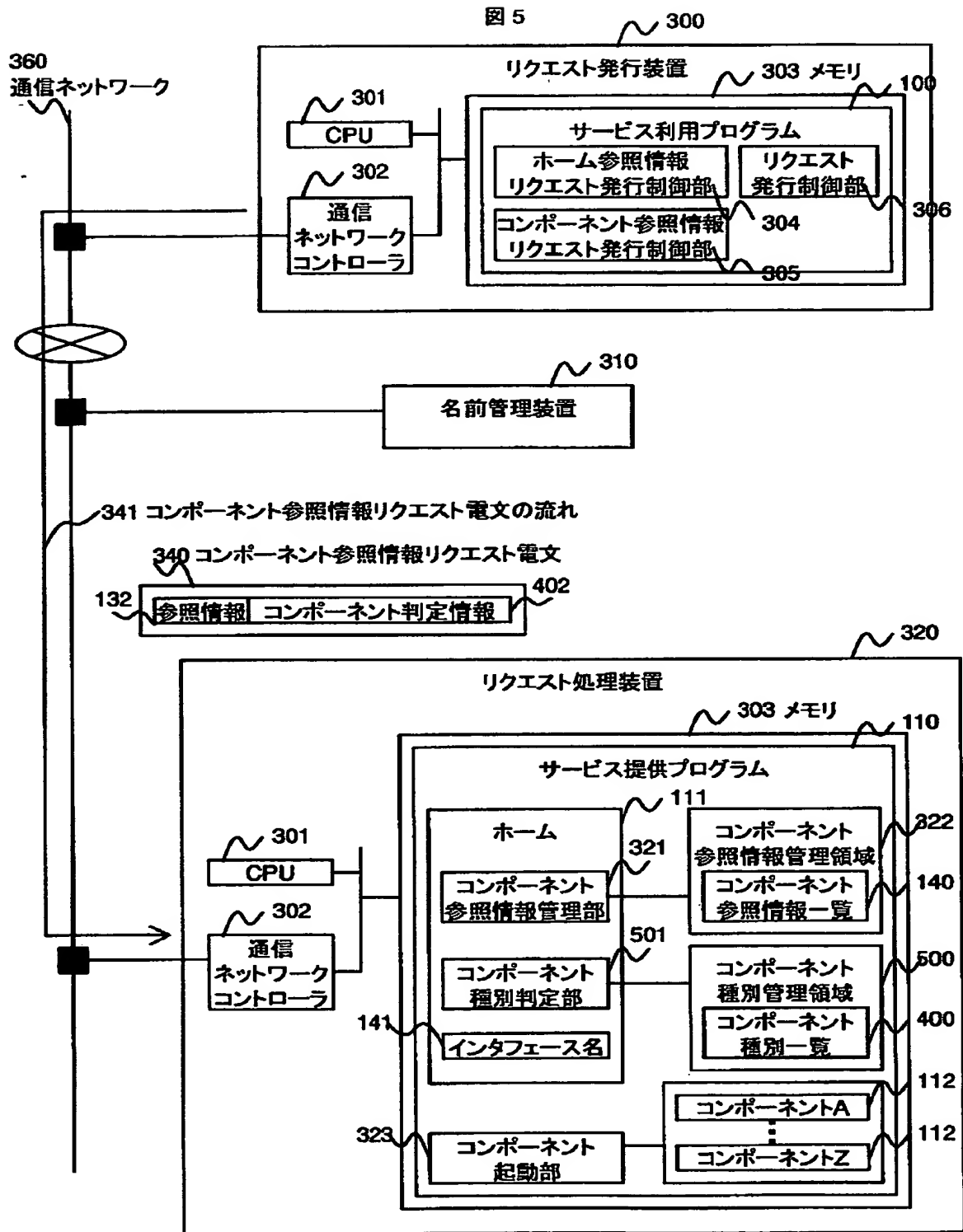


【図 4】

図 4

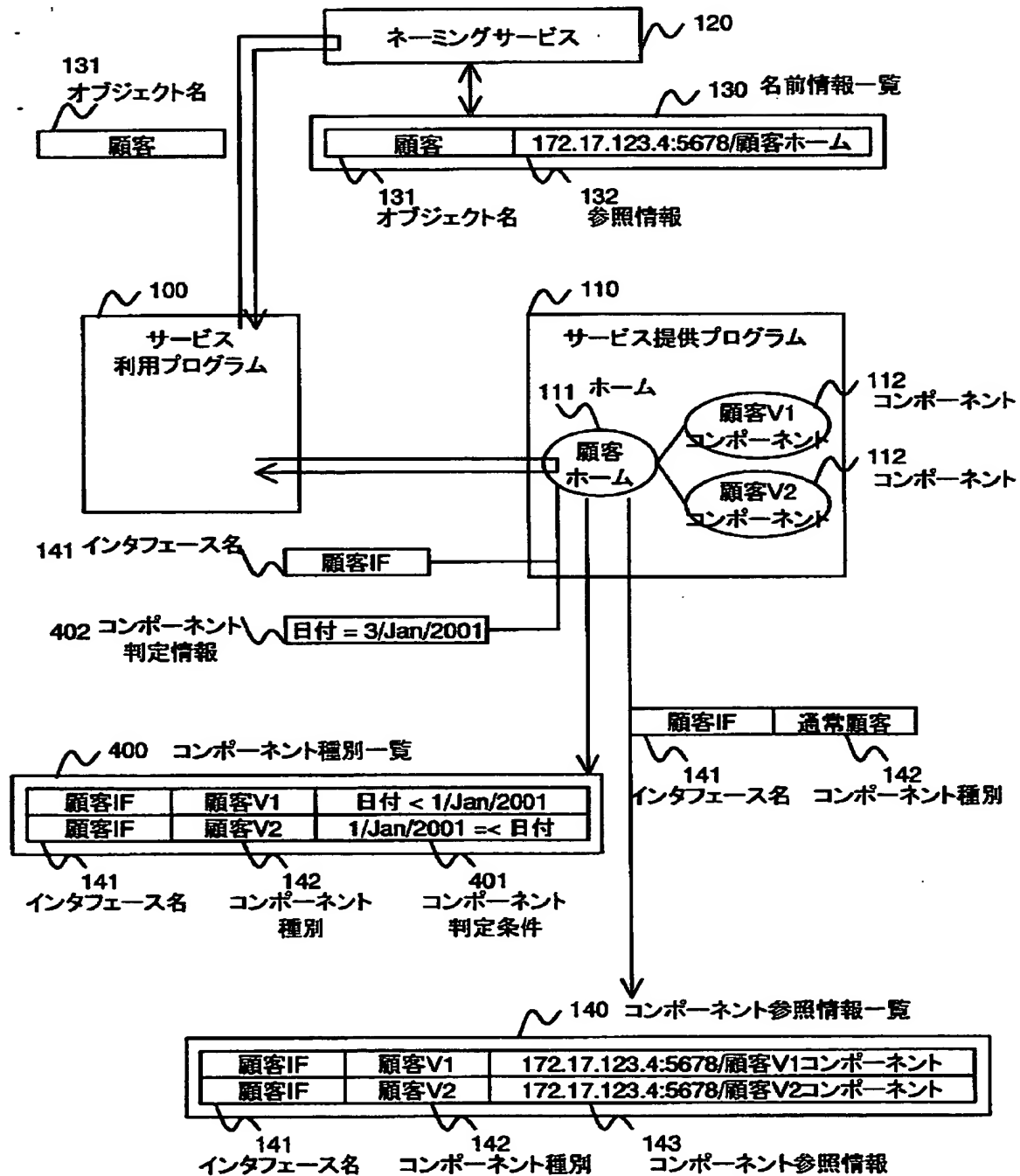


【図 5】



【図 6】

図 6



【書類名】 要約書

【要約】

【課題】

コンポーネントモデルに基づくシステムにおいて、コンポーネントの種類が増加すると、ネーミングサービスへの登録情報が増加し、運用管理コストが増大する。

【解決手段】

サービス利用プログラムは、ホームからコンポーネント参照情報を取得するとき、コンポーネント種別（例えば、通常顧客）を渡す。ホームは、一つのインタフェース名を保持し、これを満たす複数種類のコンポーネントを管理する。ホームは、コンポーネント参照情報一覧を参照し、渡されたコンポーネント種別と、保持するインタフェース名に対応するコンポーネント参照情報を返す。そして、サービス利用プログラムは、取得したコンポーネント参照情報を利用して、コンポーネント（例えば、通常顧客コンポーネント）を呼び出す。

【選択図】 図1

出 願 人 履 歴 情 報

識別番号 [000005108]

1. 変更年月日 1990年 8月31日

[変更理由] 新規登録

住 所 東京都千代田区神田駿河台4丁目6番地

氏 名 株式会社日立製作所